Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 2 (2014) 189 – 194

Transportation Research Procedia

www.elsevier.com/locate/procedia

The Conference on Pedestrian and Evacuation Dynamics 2014 (PED2014)

Time-Of-Flight technology applied in pedestrian movement detection

Minjie Chen [a,*], Günter Bärwolff [a], Hartmut Schwandt [a]

[a] *Institut für Mathematik, Technische Universität Berlin, Straße des 17. Juni 136, 10623 Berlin, Germany*

Abstract

We propose a new method based on the Time-Of-Flight (TOF) measuring technology for the automatic pedestrian trajectory extraction. We start with a new data structure called TOF-tree for the segmentation of the frames in a sequence of recordings. The tree structure enables an efficient calculation of the segmented objects (TOF-objects). While a real object (pedestrian etc.) in the scene is always associated with a set of connecting TOF-objects, combined TOF-objects can be matched with each other from frames in the sequence to reconstruct the trajectories of the moving objects in the original scene. The trajectory information of the pedestrians can be adapted for industrial usage, for example, automatic passenger counting in public transportation.

Peer-review under responsibility of Department of Transport & Planning Faculty of Civil Engineering and Geosciences
Delft University of Technology

Keywords: distance measurement; Time-Of-Flight; object detection; pedestrian movement; trajectory extraction

1. Introduction

For the validation of pedestrian simulation tools real-world data are indispensable. In particular, trajectory information of pedestrians involved in the test environment is often sought after. In the current work we propose a new method based on the Time-Of-Flight (TOF) measuring technology (Schwarte et al. (1995); Spirig et al. (1995); Lange (2000)) for the automatic object detection and trajectory extraction. In this context, distance measurement is the very first step to undertake. Traditional laser-based methods of distance measurement requires precise measurement of light travel duration. In a short distance range, this is hard to achieve even by means of contemporary technology. In TOF, the phase shift $\Delta\phi$ of an amplitude-modulated light wave with frequency f plays a key role. Within a certain distance range $d < d_{\max}$, the following relationship between the phase shift $\Delta\phi$ and the light travel duration can be established

$$\Delta\phi = \frac{4\pi f}{c} \cdot d \tag{1}$$

in measuring the distance d. Thus, direct measurement of light travel duration is no longer necessary. The maximum valid range of (1) is $d_{\max} = \frac{c}{2f}$, with c denoting the light speed as in (1). A typical modulation frequency $f = 20\text{MHz}$

* Corresponding author. Tel.: +49-30-314-28565.
E-mail address: minjie.chen@math.tu-berlin.de

Peer-review under responsibility of Department of Transport & Planning Faculty of Civil Engineering and Geosciences Delft University of Technology
doi:10.1016/j.trpro.2014.09.028

would guarantee a maximum distance of $d_{\max} = 7.5$m for the ranging system. A good explanation of the calculation of $\Delta\phi$ via sampled intensity values of the light signal can be found in Creath (1988). For a detailed error analysis of the measurement we refer to Frank et al. (2009).

2. System construction

Our system contains a TOF sensor installed perpendicular to the ground floor Ω (also called observation area) which provides us with a cloud of points at the time of recording. The positions of these points can be easily converted into (x, y, z)-coordinates in the coordinate system defined by the floor plane and the optical axis of the sensor, see Fig. 1. x and y can be rescaled into integers to denote the discrete positions in the floor plane. After resampling,

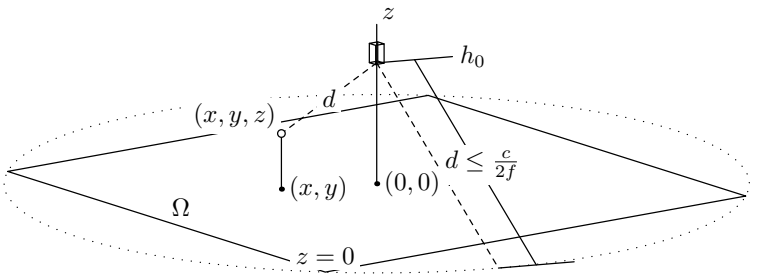

Fig. 1. Installation of the TOF sensor at a height of h_0 in the local coordinate system. The radial distance d to the object will be converted into the z-component of the coordinates.

we acquire the height information of the two-dimensional intervals associated with the (x, y)-pairs. The recording of the scene through the sensor provides us with ranging data stored in a sequence of n frames $F_0, \ldots, F_{n-1}$, and in each of these frames, a set of sample points of (x, y).

Our previous work (Chen et al. (2010)) proposed a quadtree-like data structure called "TOF-node". Each TOF-node stores the z-value associated with the position (x, y) and holds pointers to up to four child TOF-nodes (written as W, N, E and S). Naturally, the child pointers are allowed to be void and thus de-reference no further TOF-nodes. A simplified representation of this structure is given in Fig. 2. The child nodes, written as W (west, stands for $(\Delta x, \Delta y) = (-1, 0)$),

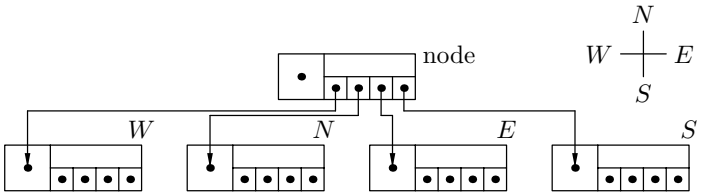

Fig. 2. Simplified representation of the TOF-node. The TOF-node "node" stands for a TOF-tree at the same time.

N (north, $(\Delta x, \Delta y) = (0, 1)$), E (east, $(\Delta x, \Delta y) = (1, 0)$) and S (south, $(\Delta x, \Delta y) = (0, -1)$), refer to the immediate neighbouring positions with $(\pm 1, 0)$ and $(0, \pm 1)$ where the z-values are lower. A TOF-node, which is itself not child of any TOF-nodes, will be referred to as a "TOF-tree" (cf. Fig. 2). Since the monotonicity respecting the z-values is imposed, the root node of a TOF-tree is always associated with a local maximum in z. Furthermore, convex objects can be represented in TOF-trees. We notice that in this data structure the original geometric information is completely preserved.

3. Algorithm

3.1. Construction of TOF-trees

Owing to the page limit of the current text, we omit the full technical specification of the procedure for the construction of TOF-trees, an earlier version of this can be found in Chen et al. (2010). The structure of TOF-tree establishes the monotonicity of the height values among the sample points in a frame. Since this construction procedure processes all the sample points in the scene, its result would be a segmentation of the sample points in the observation area Ω. Sample points with extra noises (due to measurement inaccuracy or error etc.) would lead to isolated TOF-trees. These TOF-trees have generally very small sizes and can be easily filtered out. Although this results later in holes of the reconstructed objects, the side effects of this phenomenon (for example, in retrieving geometric information of the objects) are in most cases neglectable.

3.2. Processing of TOF-trees

The main task of the processing of a TOF-tree is to retrieve its boundary information (for example, of its neighbouring TOF-trees) on an arbitrary height level. In every frame F_i ($i = 0, \ldots, n-1$), we have a segmentation of the sample points, composed of a collection of TOF-trees $t_{i,0}, \ldots, t_{i,f_i-1}$ (with f_j denoting the number of the TOF-trees constructed in the frame F_j, $j = 0, \ldots, n-1$). We observe that a geometrically convex object can be represented by a unique TOF-tree whereas the objects detected in a real scene can generally be represented by sets of connected TOF-trees. Let $T_i = \{0, \ldots, f_i - 1\}$ be the set of the indices of the TOF-trees. Let further 2^{T_i} denote the power set of T_i. For every nonempty element B of 2^{T_i} ($B \subseteq T_i$, $B \neq \emptyset$), we examine the boundary information of the TOF-trees associated with this index set B: when these are connected, we will construct a so-called "TOF-object" with these indices. A TOF-object can be thus considered as a union of the TOF-trees associated with the indices from the set B; it serves as a candidate for a real object to be detected in the scene. We notice the almost exponential complexity of this step through introducing the power set 2^{T_i}, so the pre-processing (smoothing etc.) of the original sampled data and the filtering of the irrelevant TOF-trees are very necessary. Empirically speaking, however, above a certain scale, the growth of complexity in this step is much lower than being exponential, since most elements of 2^{T_i}, as indices of the TOF-trees, are not qualified to generate TOF-objects (with the corresponding TOF-trees being not connected). The geometric information of the TOF-objects can be easily computed once we know the corresponding TOF-trees. Under this assumption it is possible to apply templates on TOF-objects for the detection of humans as special objects.

3.3. Matching of TOF-objects

The last two steps are performed within the separate frames. We now define another data structure called "TOF-trajectory".

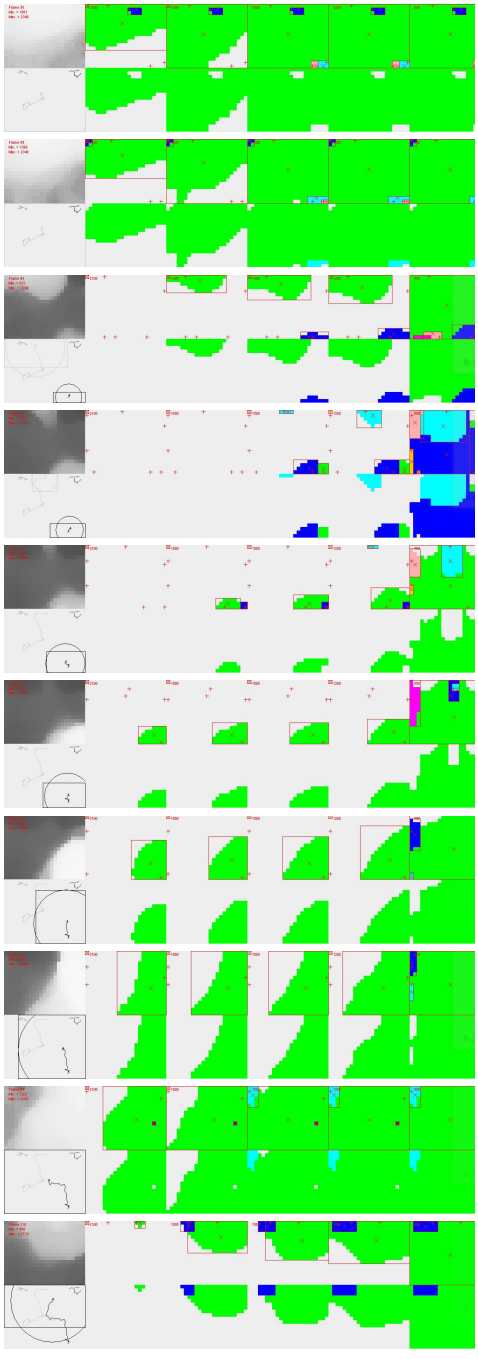

Fig. 3. Frames from a simulation example of two passengers walking through a door. Trajectories of the passengers are shown with bounding-boxes. The segmentation of the sample points is shown on different height levels, as in the coloured figures.

A TOF-trajectory is a record of TOF-objects in a continuous sequence of frames. Obviously, given such a record, the physical trajectory (or position evolution) of the corresponding real object (or human) detected in the scene can be reconstructed. In the start frame F_0, we initialize a series of incomplete TOF-trajectories $l_{0,0}, \ldots, l_{0,f_0-1}$ by the present TOF-trees in F_0. In F_1, these trajectories will be marked as active, if not otherwise closed (completed). In a following frame F_i ($i = 1, \ldots, n-1$), a TOF-trajectory $l_{i-1,k}$ will be marked as active, if $l_{i-1,k}$ contains a TOF-object (addressed by k) from the frame F_{i-1}. Assume, in frame F_i, we have constructed a collection of TOF-objects $O_i = \{o_{i,0}, \ldots, o_{i,|O_i|-1}\}$ (with obviously $|O_i| \leq 2^{|T_i|} - 1$); at the same time, we have a collection of incomplete TOF-trajectories $L_{i-1} = \{l_{i-1,0}, \ldots, l_{i-1,|L_{i-1}|-1}\}$ derived from the previous frame F_{i-1}. It is then possible to construct a distance matrix D_i with $|O_i| \cdot |L_{i-1}|$ items:

$$D_i = \left(d_{i,u,v}\right), \text{for } u = 0, \ldots, |O_i| - 1, v = 0, \ldots, |L_{i-1}| - 1, \tag{2}$$

where $d_{i,u,v}$ is the distance from the TOF-object $o_{i,u} \in O_i$ to the active TOF-trajectory $l_{i-1,v} \in L_{i-1}$. The distance function to compute $d_{i,u,v}$ in (2) can be defined independently. A very straightforward suggestion for $d_{i,u,v}$ can be:

$$d_{i,u,v} = |o_{i,u} - o_{i-1,v}|_2, \tag{3}$$

where $o_{i-1,v}$ denotes the TOF-object stored in the TOF-trajectory $l_{i-1,v}$ in the previous frame F_{i-1}. The geometric centre of a TOF-object $o_{i,u}$, which is again the weighted average of the centres of the associated TOF-trees generated by the index set B, can be used to describe its position:

$$\frac{\sum_{b \in B} t_{i,b}}{\sum_{b \in B} \left|t_{i,b}\right|}.$$

The next step is to match the current TOF-objects with the active TOF-trajectories and update the TOF-trajectories for the subsequent frame F_{i+1}:

procedure MATCH**:**
parameter: O_i, L_{i-1}

mark incomplete TOF-trajectories from L_{i-1} as active;
compute D_i and set all items in D_i as active;

repeat
select minimum $d_{i,u,v}$ from all active items in D_i;
if $d_{i,u,v}$ below threshold
add $o_{i,u}$ to $l_{i-1,v}$;
deactivate $l_{i-1,v}$;
else
construct an incomplete TOF-trajectory by $o_{i,u}$ for the next frame F_{i+1};
fi
deactivate all items $d_{i,u',v'}$ in D_i for all $u' = 0, \ldots, |O_i| - 1$, $u' \neq u$ that $o_{i,u} \cap o_{i,u'} \neq \emptyset$ and $v' = 0, \ldots, |L_{i-1}| - 1$; (∗)
until no active item in D_i left

if there are active TOF-trajectories left
// but there should be no TOF-object
mark all active TOF-trajectories as closed;
else
// there are active TOF-objects left but there should be no TOF-trajectory
repeat
construct an incomplete TOF-trajectory by an active TOF-object for the next frame F_{i+1};
deactivate all other overlapping active TOF-objects;
until no active TOF-objects left
fi

return

Procedure MATCH is composed of three parts. The first one is the initialization and computes a distance matrix D_i relating to the current TOF-objects and TOF-trajectories in the frame. The second part searches for the minimum in the distance matrix after every update of the TOF-objects and TOF-trajectories; the distance matrix D_i itself, however, stays unchanged. Given the successful search result of a TOF-object and a TOF-trajectory, we either, if the current minimum distance is below an empirical threshold value, append the TOF-object to the TOF-trajectory, or we define a new TOF-trajectory with the TOF-object. The empirical threshold value applied here needs to be calibrated with consideration of the external distance function (3) with which the distance matrix D_i is computed. Line $(*)$ says that once a TOF-object has been adopted to construct or update a TOF-trajectory, all the other TOF-objects with which there is an overlapping of the TOF-trees must be removed from the current frame F_i, since they are no longer valid candidates for the matching in the current frame. The third part of this procedure deals with the rest TOF-objects or TOF-trajectories, but not both, because if there were both TOF-objects and TOF-trajectories left, there would be also active $d_{i,u,v}$ items remaining in the distance matrix D_i. The TOF-trajectories we have here can be considered as complete, since no TOF-object will be attached to them, they will not be activated in the next frame. Similarly, the rest TOF-objects will be used to construct new TOF-trajectories, since they are not matched with any active TOF-trajectory in the current frame. Naturally, we need to run the procedure MATCH through the frames $F_1, \ldots, F_{n-1}$. Frames from an simulation example are given in Fig. 3.

4. Possible extension and further discussion

The above approach can be regarded as a very simple contour-based tracking method. In combination with templates for geometric form recognition this method can be deployed for the purpose of pedestrian detection and automatic trajectory extraction. The background idea of our method is to register detected objects (pedestrians) in the scene with their possible contour lines and then apply a simple matching of position transition on the registered contours. The generation of contour lines is very different from traditional methods, owing to the specific nature of the data delivered by the TOF ranging system. Since ours is a very rudimentary approach in the processing of TOF data, we have only applied a very simple matching method to establish the relationship of an object's actual position to its possible trace over the time (that is, trajectory). The distance function $d_{i,u,v}$ in the procedure MATCH is designed deliberately to be an external function for further extension. For example, to compute the distance matrix (2), (3) can be generalized to include a position prediction under the assumption of a constant velocity locally:

$$d_{i,u,v} = |o_{i,u} - o'_{i,v}|_2, \tag{4}$$

with $o'_{i,v}$ denoting the position prediction of the TOF-object stored in the TOF-trajectory $l_{i-1,v}$ in the current frame F_i. With a constant frame frequency we will have:

$$o'_{i,v} = 2o_{i-1,v} - o_{i-2,v}.$$

(4) implies that that procedure MATCH requires the information of another frame F_{i-2} for the processing of the current frame F_i in the data sequence. Apart from this extension, additional matching methods can be embedded as function modules as well.

Since modulation-based TOF has a comparatively short range limit (cf. (1)), this method is applicable primarily in small observation areas. However, it is still applicable in various contexts, for example, in the automatic passenger counting in public transportation. Our method itself is hardware-independent. The specific TOF hardware we used is a SwissRanger4000 sensor[1] with a resolution of 176×144. We had conducted a series of tests of passenger counting on the real-world data collected by a local public transportation company in Berlin. Our system showed a slightly improved performance against the current industrial counting system in use, see Fig. 4 for a comparison. Obviously our system as merely a prototype will require many efforts in further development, especially in the task of pedestrian (object) detection, but basically the TOF technology offers a new idea for the solution of similar problems. In comparison to SwissRanger4000's rather high resolution, even a much lower resolution would be acceptable in the

[1] Product of Mesa Imaging AG (Switzerland), homepage `http://www.mesa-imaging.ch`.

practice, if the range information is acquired with a sufficient precision (so that the geometric information, especially shape, of the objects can be reconstructed). However, the quality of data collection depends on further factors like light conditions which are generally discussed in the context of optical ranging systems.

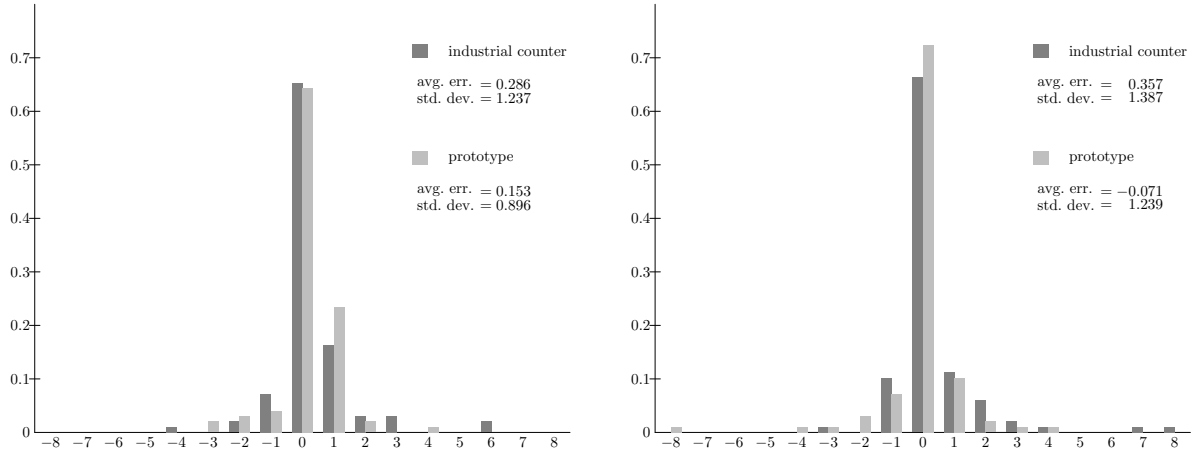

Fig. 4. Error statistics of the counting in two directions. Horizontal axis stands for the difference to the correct counting result; vertical axis the distribution of errors (the bar lengths refer to the relative error occurrences of the corresponding counter). Average error and standard deviation of the counters are given.

Acknowledgement

The authors gratefully acknowledge the support of Federal Ministry for Economic Affairs and Energy of Germany for the project VP2653402RR1 and federal state of Berlin/Investitionsbank Berlin for the project 10153525.

References

Chen, M.-J., Plaue, M., Bärwolff, G., Schwandt, H., 2010. The tree structure for Time-Of-Flight dataset representation, An electronic version (preprint) can be retrieved at: `http://page.math.tu-berlin.de/~chenmin/pub/cpbs100318.pdf` (accessed June 30, 2014).

Creath, K., 1988. Phase-measurement interferometry techniques. *Progress in Optics*, 26:349–393.

Frank, M., Plaue, M., Rapp, H., Köthe, U., Jähne, B., Hamprecht, F. A., 2009. Theoretical and experimental error analysis of continuous-wave time-of-flight range cameras. *Optical Engineering*, 48(1):013602.

Lange, R., 2000. *3D Time-of-flight distance measurement with custom solid-state image sensors in CMOS/CCD-technology*. PhD thesis, Universität Siegen.

Schwarte, R., Heinol, H.-G., Xu, Z., Hartmann, K., 1995. New active 3D vision system based on rf-modulation interferometry of incoherent light. *Proc. SPIE*, 2588, Intelligent Robots and Computer Vision XIV: Algorithms, Techniques, Active Vision, and Materials Handling:126–134.

Spirig, T., Seitz, P., Vietze, O., Heitger, F., 1995. The lock-in CCD—two-dimensional synchronous detection of light. *IEEE Journal of Quantum Electronics*, 31(9):1705–1708.